Protogate Freeway® Security Features User's Guide (SFUG)

DC 908-3004A

Protogate, Inc.

12225 World Trade Drive
Suite R
San Diego, CA
92128
USA
Web: www.protogate.com
Email: sales@protogate.com
Voice: (858) 451-0865
Fax: (877) 473-0190

Protogate Freeway® Security Features User's Guide (SFUG): DC 908-3004A

by Protogate, Inc.

Published October 2013

Copyright © 2013 Protogate, Inc.

This Freeway® Security Features User's Guide (SFUG) document describes the components of the Protogate Freeway software which can be used to enhance security.

The contents of this document are considered to be sensitive, and dissemination of this document should be restricted; only those who are interested in improving the security level of Protogate Freeway systems should read this document.

This document can change without notice. Protogate, Inc. accepts no liability for any errors this document might contain.

Freeway is a registered trademark of Protogate, Inc. All other trademarks and trade names are the properties of their respective holders.

Table of Contents

Preface ..... **v**
Purpose of Document v
Intended Audience v
Organization of Document v
Protogate References vi
Document Conventions viii
Revision History viii
Customer Support ix
1. Scope **10**
1.1. Identification 10
1.2. System Overview 10
1.3. Document Overview 10
2. Reference Documents **11**
3. Logs **12**
3.1. Configuring Logging 12
3.2. Maintaining Logs 12
4. Firewall **14**
4.1. Configuring the Firewall 14
5. Auditing **15**
5.1. Configuring Auditing 15
5.2. Maintaining Audit Trail Logs 16
6. Hardening a Freeway **18**
6.1. Freeway Firewall Settings 18
6.2. Unnecessary Services 19
6.3. Disallow Direct Root Login 19
6.4. NTP (Network Time Protocol) 20
6.5. SNMP (Simple Network Management Protocol) 20
6.6. Secure the Webserver 21
6.7. Enable Logging 22
6.8. Rotate Log Files 22
6.9. Configure Auditing 24
7. Notes **27**
A. Sample rc.startsra File **28**
A.1. `rc.startsra` Configuration File 28
Index **37**
Colophon **39**

List of Tables

1. Revision History viii
2-1. Referenced Documents..... 11
7-1. Acronym definitions 27

Preface

Purpose of Document

This Freeway® Security Features User's Guide (SFUG) document identifies the capabilities of a Freeway which can be used to enhance security.

Intended Audience

The contents of this document are considered to be sensitive, and dissemination of this document should be restricted; only those who are interested in improving the security level of Protogate Freeway systems should read this document. This document is intended for system administrators who want a better understanding of how to configure a Protogate Freeway to be more secure.

Organization of Document

This document is organized into the following major sections:

Chapter 1

is an overview of this document and of the Protogate Freeway software.

Chapter 2

is a list of other documents referenced by this document.

Chapter 3

describes the Freeway logging capabilities.

Chapter 4

describes the Freeway firewall capabilities.

Chapter 5

describes how to setup and use auditing on the Freeway.

Chapter 6

describes how to tighten the security on ("harden") a Freeway.

Chapter 7

includes general information to aid in understanding this document.

Appendix A

includes a sample rc.startsra file, to show how to configure and enable many of the security features described in this document.

Protogate References

The following general product documentation list is provided to familiarize you with the available Protogate Freeway and embedded ICP products. Most of these documents are available on-line at Protogate's website (http://www.protogate.com/). Additional information about documents which are specifically referenced by this Freeway Security Features User's Guide (SFUG) document are in Chapter 2 of this document.

General Product Overview Documents

Freeway 1100 Technical Overview	25-000-0419
Freeway 2000/4000/8800 Technical Overview	25-000-0374
ICP2432 Technical Overview	25-000-0420
ICP6000X Technical Overview	25-000-0522

Hardware Support Documents

Freeway 500 Hardware Installation Guide	DC-900-2000
Freeway 1100/1150 Hardware Installation Guide	DC-900-1370
Freeway 1200/1300 Hardware Installation Guide	DC-900-1537
Freeway 2000/4000 Hardware Installation Guide	DC-900-1331
Freeway 8800 Hardware Installation Guide	DC-900-1553
Freeway 3100 Hardware Installation Guide	DC-900-2002
Freeway 3200 Hardware Installation Guide	DC-900-2003
Freeway 3400 Hardware Installation Guide	DC-900-2004
Freeway 3600 Hardware Installation Guide	DC-900-2005
Freeway 3110 Hardware Installation Guide	DC-900-2012
Freeway 3210 Hardware Installation Guide	DC-900-2013
Freeway 3410 Hardware Installation Guide	DC-900-2014
Freeway 3610 Hardware Installation Guide	DC-900-2015
Freeway 3112 Hardware Installation Guide	DC-900-2016
Freeway 3212 Hardware Installation Guide	DC-900-2017
Freeway 3412 Hardware Installation Guide	DC-900-2018
Freeway 3612 Hardware Installation Guide	DC-900-2019
Freeway ICP6000R/ICP6000X Hardware Description	DC-900-1020
ICP6000(X)/ICP9000(X) Hardware Description and Theory of Operation	DC-900-0408
ICP2424 Hardware Description and Theory of Operation	DC-900-1328

ICP2432 Hardware Description and Theory of Operation	DC-900-1501
ICP2432 Electrical Interfaces (Addendum to DC-900-1501)	DC-900-1566
ICP2432 Hardware Installation Guide	DC-900-1502
ICP2432B Hardware Installation Guide	DC-900-2009

Freeway Software Installation and Configuration Support Documents

Freeway User Guide	DC-900-1333
Freeway Loopback Test Procedures	DC-900-1533
Freeway Release Addendum: Client Platforms	DC-900-1555
Freeway Message Switch User Guide	DC-900-1588
Freeway Software Requirements Specification (SRS)	DC-900-2021
Freeway Ports, Protocols, and Services (PPS)	DC-900-2022
Freeway Software Version Description (SVD)	DC-900-2023
Freeway Lifecycle Support Plan (LSP)	DC-900-2024
Freeway Security Features User's Guide (SFUG)	DC-908-3004
Freeway Security Target (ST)	DC-908-3005

Embedded ICP Software Installation and Programming Support Documents

ICP2432 User Guide for Digital UNIX	DC-900-1513
ICP2432 User Guide for OpenVMS Alpha	DC-900-1511
ICP2432 User Guide for OpenVMS Alpha (DLITE Interface)	DC-900-1516
ICP2432 User Guide for Solaris STREAMS	DC-900-1512
ICP2432 User Guide for Windows NT	DC-900-1510
ICP2432 User Guide for Windows NT (DLITE Interface)	DC-900-1514

Application Program Interface (API) Programming Support Documents

Freeway Data Link Interface Reference Guide	DC-900-1385
Freeway Transport Subsystem Interface Reference Guide	DC-900-1386
QIO/SQIO API Reference Guide	DC-900-1355

Socket Interface Programming Support Documents

Freeway Client-Server Interface Control Document	DC-900-1303

Toolkit Programming Support Documents

Freeway Server-Resident Application (SRA) Programmer Guide	DC-900-1325

OS/Impact Programmer Guide	DC-900-1030
Freeway OS/Protogate Programmer's Guide	DC-900-2008
Protocol Software Toolkit Programmer Guide	DC-900-1338
Protocol Software Toolkit Programmer's Guide (ICP2432B)	DC-900-2007

Protocol Support Documents

ADCCP NRM Programmer Guide	DC-900-1317
Asynchronous Wire Service (AWS) Programmer Guide	DC-900-1324
AUTODIN Programmer Guide	DC-908-1558
Bit-Stream Protocol Programmer Guide	DC-900-1574
BSC Programmer Guide	DC-900-1340
BSCDEMO User Guide	DC-900-1349
BSCTRAN Programmer Guide	DC-900-1406
DDCMP Programmer Guide	DC-900-1343
FMP Programmer Guide	DC-900-1339
Military/Government Protocols Programmer Guide	DC-900-1602
N/SP-STD-1200B Programmer Guide	DC-908-1359
NASCOM Programmer's Guide	DC-900-2010
SIO STD-1300 Programmer Guide	DC-908-1559
TIMI Programmer's Guide	DC-900-2011
X.25 Call Service API Guide	DC-900-1392
X.25/HDLC Configuration Guide	DC-900-1345
X.25 Low-Level Interface	DC-900-1307

Document Conventions

In this document, the term "Freeway" refers generically to all current Protogate Freeway models; for example: the Freeway 3112, the Freeway 3212, the Freeway 3412, and the Freeway 3612.

Revision History

The revision history of the Freeway Security Features User's Guide (SFUG), Protogate document DC 908-3004, is recorded below:

Table 1. Revision History

Revision	Release Date	Description
DC 908-3004A	October, 2013	Initial Release

Customer Support

If you are having trouble with any Protogate product, call us at 1-858-451-0865 (U.S.) Monday through Friday between 8 a.m. and 5 p.m. Pacific time. You can also fax your questions to us at (858) 451-2865 or (877) 473-0190 any time. Please include a cover sheet addressed to "Customer Service." We are always interested in suggestions for improving our products. You can use the report form in the back of this manual to send us your recommendations.

Chapter 1. Scope

1.1. Identification

This document describes the capabilities of a Protogate Freeway® which can be used to enhance security.

1.2. System Overview

The Protogate Freeway is a data communication system which connects one or more serial-link channels (Wide-Area-Network, or WAN channels) of various types to one or more IP (Internet Protocol) networks. The Freeway acts as a gateway, providing WAN channel access to clients on the IP network.

All Protogate Freeways run custom-built software which is written and provided by Protogate, and which completely controls the Freeway. The Freeway software is based on a version of the FreeBSD operating system which has been modified to control one or more Protogate Intelligent Communications Processor (ICP) boards. ICP boards are Protogate-manufactured boards which can be installed into a Freeway chassis, plugged into one or more serial-link channels, and configured to implement a data communications protocol. Each ICP board installed into a Freeway provides 2, 4, or 8 WAN ports.

1.3. Document Overview

This document describes some of the security capabilities of a Protogate Freeway . The contents of this document are considered to be sensitive, and dissemination of this document should be restricted; only those who are interested in improving the security level of Protogate Freeway systems should read this document.

Chapter 2. Reference Documents

A full list of Protogate documents is in the Preface Section of this document.

Documents referenced by this Freeway Security Features User's Guide (SFUG) document are listed in Table 2-1.

Table 2-1. Referenced Documents

Number	Title	Revision	Date
DI-MCCR-81349	Data Item Description (DID): Security Features User's Guide (SFUG)		02 Jul, 1993
DI-MCCR-81857	Data Item Description (DID): System Security Administrator Operators Documentation (SSAOD)		21 Dec, 2011
DC-900-1333	Freeway User's Guide	Q	Sep, 2013
DC-900-2016	Freeway 3112 Hardware Installation Guide	A	Sep, 2011
DC-900-2017	Freeway 3212 Hardware Installation Guide	A	Sep, 2011
DC-900-2018	Freeway 3412 Hardware Installation Guide	A	Sep, 2011
DC-900-2019	Freeway 3612 Hardware Installation Guide	A	Sep, 2011

Chapter 3. Logs

This chapter describes how to setup and use logging on a Freeway . Logging is useful to keep security high because it allows a system administrator to examine the past history of the Freeway system, to see if the Freeway is being attacked or used in an inappropriate way, if some errors have occurred or resources are being over-utilized, which users have logged in, etc.

3.1. Configuring Logging

The Freeway uses the `syslogd` daemon to log system events, and the `/etc/syslog.conf` configuration file to control logging. As usual with Freeway configuration file changes, the best and most flexible way to configure logging is to put the changes into one centrally-located place, such as the file `/usr/local/freeway/boot.src/rc.startsra`, so that changes can be controlled and preserved when the overall Freeway software is upgraded to a new version. That means that the actual line changes necessary to enable the `syslogd` daemon must be inserted into the appropriate files by commands in `/usr/local/freeway/boot.src/rc.startsra`.

For example, Figure 3-1 shows lines which could be added to a Freeway's `/usr/local/freeway/boot.src/rc.startsra` file to configure and enable the `syslogd` daemon:

Figure 3-1. Configure and Enable the syslogd Daemon

```
export LOG_DIR="/var/log"
touch ${LOG_DIR}/all.log

echo "*.*           ${LOG_DIR}/all.log"         > /etc/syslog.conf

if [ -x /usr/sbin/syslogd ]; then
   /usr/sbin/syslogd
fi
```

That example would cause all loggable events to be written to the file `/var/log/all.log`. Because that file would continue to grow larger, and would eventually fill the filesystem where it exists, rotating the logs is also important. Section 3.2 shows lines how to configure a Freeway to automatically rotate and maintain the log files, to archive them and prevent them from filling a filesystem.

Another `syslog` configuration example is in Section 6.7. More information about logging and syslog configuration is available by logging into a Freeway with any user account and typing `man syslog`, `man syslog.conf`, or `man syslogd`.

3.2. Maintaining Logs

To setup a Freeway to maintain the logs files automatically, archiving old copies, compressing them if desired, and deleting the oldest log files when necessary to prevent filling a filesystem, the Freeway can use the `newsyslog` utility. That utility is controlled and configured by the `newsyslog.conf` file, and can be run automatically

(generally once per day) by the `cron` daemon. Figure 3-2 shows lines which could be added to a Freeway's `/usr/local/freeway/boot.src/rc.startsra` file to rotate and maintain the `syslog` logs:

Figure 3-2. Rotate the Log Files

```
echo "#   logfilename            mode count size  when flags"  > /etc/newsyslog.conf
echo "${LOG_DIR}/all.log          600  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/cron             600  31     *   @T05  WZ"   >> /etc/newsyslog.conf


echo "SHELL=/bin/sh"                                          > /etc/crontab
echo "PATH=/etc:/bin:/sbin:/usr/bin:/usr/sbin"               >> /etc/crontab
echo "HOME=/var/log"                                         >> /etc/crontab
echo "B_FWY_SERVERNAME=${B_FWY_SERVERNAME}"                  >> /etc/crontab
echo "MAILTO=\"\""                                           >> /etc/crontab
echo "TZ=\"\""                                               >> /etc/crontab
echo "#"                                                     >> /etc/crontab
echo "#minute hour   mday   month   wday   who     command"   >> /etc/crontab
echo "#"                                                     >> /etc/crontab
echo "0        *      *      *       *      root   newsyslog" >> /etc/crontab


if [ -x /usr/sbin/cron ]; then
   /usr/sbin/cron
fi
```

Those lines create two new files, `/etc/newsyslog.conf` and `/etc/crontab`, to cause the `cron` daemon to run the `newsyslog` utility every night at midnight. As configured there, `newsyslog` would compress the existing `/var/log/all.log` file and move it to a new name (`/var/log/all.log.0.gz`), then create a new, initially empty, `/var/log/all.log`file to be filled with new syslog entries. It will preserve up to 31 previous copies of `all.log.` (from `all.log.0.gz` to `all.log.30.gz`) and will delete copies older than that.

Another `newsyslog` configuration example is in Section 6.8. More information about rotating the logs and about configuring `newsyslog` is available by logging into a Freeway and typing the commands `man newsyslog`, `man newsyslog.conf`, `man cron`, and `man 5 crontab`.

Chapter 4. Firewall

This chapter describes how to setup and use the firewall on a Freeway . The firewall can protect the Freeway from unwanted connections, either by service (IP port number) or by source (IP address), or any combination of those.

4.1. Configuring the Firewall

The Freeway uses the `ipfw` daemon to configure and control the firewall. As usual with Freeway configuration, the best and most flexible way to configure the firewall is to put the desired commands into one centrally-located place, such as the file `/usr/local/freeway/boot.src/rc.startsra`, so that changes can be controlled and preserved when the overall Freeway software is upgraded to a new version.

For example, Figure 4-1 shows lines which could be added to a Freeway's `/usr/local/freeway/boot.src/rc.startsra` file to configure and enable the `ipfw` firewall:

Figure 4-1. Configure and Enable the ipfw Firewall

```
ipfw add 20100 deny ip from not me to not me
ipfw add 20200 deny icmp from me to any icmptypes 11

            # For security: deny all access to timestamp info via ICMP packets
ipfw add 20300 deny icmp from any to any icmptypes 13
ipfw add 20400 deny icmp from any to any icmptypes 14

            # For security: deny all access to unnecessary ports
ipfw add 21000 deny tcp from any to me 23

ipfw add 31000 deny tcp from 192.168.1.100 to me
```

That example would deny all attempts to send an IP packet through the Freeway if the Freeway is not either the source or destination of the packet; it would deny all ICMP type 11 packets from being sent by the Freeway; it would deny all ICMP type 13 or 14 packets from being received or sent by the Freeway; it would deny all TCP/IP packets sent to port 23 (the telnet port) of the Freeway; and it would deny all packets of any kind from the IP address 192.168.1.100.

The current firewall rules for any Freeway can be displayed by logging into the Freeway, using the `su - shell` to gain root (shell) access rights, and then running the command: `ipfw show`.

Another example set of `ipfw` commands is shown in Section 6.1. More information about ipfw and firewall configuration is available by logging into a Freeway with any user account and typing `man ipfw`.

Chapter 5. Auditing

This chapter describes how to setup and use system-level event auditing on a Freeway. Auditing is useful to keep security high because it allows a system administrator to examine the past history of the Freeway system, to see which users have logged in and exactly what they have been doing, in complete detail, all the way down to what system calls they have made and which files they have accessed.

5.1. Configuring Auditing

The Freeway uses the `auditd` daemon to record system-level events, and the configuration files in the `/etc/security/` directory to control system-level event auditing. As usual with Freeway configuration file changes, the best and most flexible way to configure system-level event auditing is to put the changes into one centrally-located place, such as the file `/usr/local/freeway/boot.src/rc.startsra`, so that changes can be controlled and preserved when the overall Freeway software is upgraded to a new version. That means that the actual line changes necessary to enable the `auditd` daemon must be inserted into the appropriate files by commands in `/usr/local/freeway/boot.src/rc.startsra`.

For example, Figure 5-1 shows lines which could be added to a Freeway's `/usr/local/freeway/boot.src/rc.startsra` file to configure and enable the `auditd` daemon:

Figure 5-1. Configure and Enable System-Level Event Auditing

```
if [ ! -d /var/audit ]; then
  mkdir -p -m 750 /var/audit
fi
chmod go-w /etc/security
if /usr/bin/grep "^host:" /etc/security/audit_control >/dev/null; then
  echo "host line already in audit file -- will not tamper with it..."
else
  echo "host:${B_FWY_SERVERNAME}"  >> /etc/security/audit_control
fi

                  # If audit_user file has not been altered by any user, then
                  # add default settings for the 3 initial login accounts.
if [ 5 = ‘cat /etc/security/audit_user |wc -l‘ ]; then
  echo "#"                                                              >> /etc/security/audit_user
  echo "# These lines have been added to this file by the"              >> /etc/security/audit_user
  echo "# /usr/local/freeway/boot.src/rc.startsra command script,"     >> /etc/security/audit_user
  echo "# to configure auditing of the 3 originally-configured"        >> /etc/security/audit_user
  echo "# Freeway Monitor users.  To alter these settings, you should add"  >> /etc/security/audit_user
  echo "# echo statements in /usr/local/freeway/boot.src/rc.startsra.local," >> /etc/security/audit_user
  echo "# rather than edit either /etc/security/audit_user or"         >> /etc/security/audit_user
  echo "# /ro/etc/security/audit_user directly; using echo statements" >> /etc/security/audit_user
  echo "# will ensure that your changes are not lost or altered"       >> /etc/security/audit_user
  echo "# by any subsequent Freeway or Monitor software install."      >> /etc/security/audit_user
  echo "# See the echo statements near the end of"                     >> /etc/security/audit_user
  echo "# /usr/local/freeway/boot.src/rc.startsra for examples."       >> /etc/security/audit_user
  echo "#"                                                            >> /etc/security/audit_user
  echo "# All users which should be audited must be added here."       >> /etc/security/audit_user
  echo "#"                                                            >> /etc/security/audit_user
  echo "user:ex,ap,aa,lo,ad,na,fm,fd,fc,fw,-fr:no"                    >> /etc/security/audit_user
  echo "freeway:ex,ap,aa,lo,ad,na,fm,fd,fc,fw,-fr:no"                 >> /etc/security/audit_user
```

```
  echo "#"                                                    >> /etc/security/audit_user
fi

            # Start the kernel-level audit daemon.
/usr/sbin/auditd

            # Add posixrules file to prevent creating unnecessary audit records
if [ -f /usr/share/zoneinfo/posixrules ]; then
   echo "posixrules file exists."
else
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
   mkdir /usr/share/zoneinfo
   chmod 755 /usr/share/zoneinfo
   touch /usr/share/zoneinfo/posixrules
   chmod 444 /usr/share/zoneinfo/posixrules
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi
```

That example would cause all loggable system-level events to be written to a file in the `/var/audit/` directory. For a description of the format of the `/etc/security/audit_user` file, run `man audit_user`. For the available event types, see the `/etc/security/audit_class` and `/etc/security/audit_event` files.

Once auditing is running, the root or shell user can use a command like `praudit -l /var/audit/current` to see the audit entries, or `praudit -l /dev/auditpipe` to continually see the latest entries as they appear.

Because that `/var/audit/` directory would continue to fill with system-level event audit records, and would eventually fill the filesystem where it exists, removing and archiving the audit logs is also important. Section 5.2 shows how to configure a Freeway to automatically maintain the audit log files, to archive them and prevent them from filling a filesystem.

Another `auditd` configuration example is in Section 6.9. More information about auditing and `auditd` configuration is available by logging into a Freeway with any user account and typing any of these commands: `man audit`, `man auditd`, `man audit_class`, `man audit_event`, `man audit_user`, or `man praudit`.

5.2. Maintaining Audit Trail Logs

To setup a Freeway to maintain the audit files automatically, compressing and archiving each file when it becomes full, the Freeway can use the `audit_warn` capability. That capability is controlled and configured by the `/etc/security/audit_warn` file, and will be run automatically whenever the current system-level audit file becomes full. Figure 5-2 shows lines which could be added to a Freeway's `/usr/local/freeway/boot.src/rc.startsra` file to archive and maintain the `audit` logs:

Figure 5-2. Archive the Audit Files

```
if [ 6 = ‘cat /etc/security/audit_warn |wc -l‘ ]; then

  echo "#"                                                  >> /etc/security/audit_warn
```

```
  echo "# Added by /usr/local/freeway/boot.src/rc.startsra:"      >> /etc/security/audit_warn
  echo "#"                                                       >> /etc/security/audit_warn
  echo "# Compress and move audit trail files when they are full." >> /etc/security/audit_warn
  echo "#"                                                       >> /etc/security/audit_warn
  echo "export DATEDIR=\"\‘date -u -v ’-5S’ ’+%Y%m%d’\‘\""       >> /etc/security/audit_warn
  echo "if [ \"\$1\" = closefile ]; then"                        >> /etc/security/audit_warn
  echo "   /usr/bin/touch        /var/save/\${DATEDIR}.audit_records.zip" >> /etc/security/audit_warn
  echo "   /sbin/chown root:audit /var/save/\${DATEDIR}.audit_records.zip" >> /etc/security/audit_warn
  echo "   /sbin/chmod 600       /var/save/\${DATEDIR}.audit_records.zip" >> /etc/security/audit_warn
  echo "   /usr/local/bin/zip -r  /var/save/\${DATEDIR}.audit_records.zip \$2" >> /etc/security/audit_warn
  echo "   /usr/bin/touch        \$2.txt"                         >> /etc/security/audit_warn
  echo "   /sbin/chown root:audit \$2.txt"                        >> /etc/security/audit_warn
  echo "   /sbin/chmod 600       \$2.txt"                         >> /etc/security/audit_warn
  echo "   /usr/sbin/praudit -d ’|’ \$2 > \$2.txt"                >> /etc/security/audit_warn
  echo "   /sbin/chmod 400       \$2.txt"                         >> /etc/security/audit_warn
  echo "   /usr/local/bin/zip -r  /var/save/\${DATEDIR}.audit_records.zip \$2.txt" \
                                                                  >> /etc/security/audit_warn
  echo "   /sbin/rm -f \$2.txt"                                   >> /etc/security/audit_warn
  echo "fi"                                                      >> /etc/security/audit_warn

fi

/usr/sbin/audit -n
if /usr/bin/grep -- "^[^#]*bin\/audit -n" /etc/crontab >/dev/null; then
   echo "Audit file refresh command exists -- will not add again..."
else
   echo "0        0      *      *       *      root   /usr/sbin/audit -n" >> /etc/crontab
fi
```

Those lines adjust two files `/etc/security/audit_warn` and `/etc/crontab`. The `audit_warn` file is run whenever the audit trail file becomes full, and the new instructions within it compress the audit trail entries and move the compressed results to another file. The `crontab` change adds a line to force the `audit` program to close and reset the audit file with every reboot, and every midnight.

Another `audit_warn` configuration example is in Section 6.9. More information about archiving the system-level event audit logs is available by logging into a Freeway and typing the commands `man audit_warn`.

Chapter 6. Hardening a Freeway

This chapter shows how to use some of the capabilities of a Freeway to increase security. Many of the examples are taken from the sample `rc.startsra` file shown in (Section A.1.

The techniques used to enhance security can be grouped into these categories:

- `sysctl` settings (Section 6.1 below)
- Firewall settings (Section 6.1 below)
- Turn off unnecessary services (Section 6.2 and Section 6.3)
- Setup NTP (Network Time Daemon) (Section 6.4)
- Disable SNMP (Simple Network Management Protocol) (Section 6.5)
- Secure the webserver (Section 6.6)
- Setup system logging (Section 6.7 and Section 6.8)
- Setup system-event auditing (Section 6.9)

6.1. Freeway Firewall Settings

Figure 6-1 is an example section of a Freeway `rc.startsra` file which sets some `sysctl` specifiers and creates some firewall rules to enhance the Freeway security.

Figure 6-1. Freeway Firewall Settings

```
                              # For security: disable all pkt forwarding,
                              # using both sysctl and the firewall;
                              # disallow sending or receiving of any packet
                              # which is not sourced or destined directly
                              # to this Freeway;
                              # also disallow sending of ICMP "time exceeded"
                              # packets, which could be used by traceroute
                              # to discover information about the network.
sysctl net.inet.ip.forwarding=0
sysctl net.inet6.ip6.forwarding=0
ipfw add 20100 deny ip from not me to not me
ipfw add 20200 deny icmp from me to any icmptypes 11

            # For security: deny all access to timestamp info via ICMP packets
ipfw add 20300 deny icmp from any to any icmptypes 13
ipfw add 20400 deny icmp from any to any icmptypes 14

            # For security: deny all access to unnecessary ports
ipfw add 21000 deny tcp from any to me 23    # telnet
ipfw add 21100 deny tcp from any to me 80    # http
ipfw add 21200 deny tcp from any to me 513   # login
ipfw add 21300 deny tcp from any to me 514   # shell
# ipfw add 21400 deny tcp from any to me 20    # ftp-data
# ipfw add 21500 deny tcp from any to me 21    # ftp
# ipfw add 21600 deny tcp from any to me 8208  # Freeway daemon
```

6.2. Unnecessary Services

One of the simplest ways to enhance security on a Freeway is to turn off (disallow) all services which are not required. Access methods such as telnet and rlogin are never required on any Freeway, since users can always login via the secure shell (SSH), which provides all of the same capabilities as `telnet` or `rlogin`, but is more secure.

`telnet` and `rlogin` can be disallowed with firewall rules, as shown in Figure 6-1, but they should also be prevented from running at the daemon level, in case the firewall rules are changed or the firewall is disabled. Figure 6-2 illustrates how to disable `telnet` and `rlogin` at the daemon level. It works by commenting out the `telnet` and `login` lines in /etc/inetd.conf, so that those daemons are never run, even if a client tries to connect on their TCP/IP ports (23 is the TCP/IP port for `telnet`, and 513 is the TCP/IP port for `rlogin`).

Figure 6-2. Turning Off Unnecessary Services

```
            # For security: Turn telnet and rlogin off in inetd.conf, too
if [ -f /read_only_mounts ]; then
   mount -u -o rw  / 2>/dev/null
fi
mv /ro/etc/inetd.conf /ro/etc/inetd.conf.prev
sed -e "s/^t/#t/g" /ro/etc/inetd.conf.prev |sed -e "s/^l/#l/g" > /ro/etc/inetd.conf
if [ -f /read_only_mounts ]; then
   mount -u -o ro  / 2>/dev/null
fi
```

6.3. Disallow Direct Root Login

The root or shell accounts should not be accessible directly to users logging in across the net. Users who want to use root or shell privileges should first login under their own account, then use the `su` command to acquire root/shell privileges. This ensures that auditing works correctly, since the audit records for all actions that users take with root privileges will still be recorded under their original login account name.

Figure 6-3 shows how to disable direct root (or shell) account logins from across the net. It also creates a sample banner, to warn users who login that they are logging in to an "authorized-users-only" system.

Figure 6-3. Disallowing Direct Root Logins

```
              # For security: Disallow direct root or shell login via ssh
if /usr/bin/grep -- "^[^#]*PermitRootLogin no" /ro/etc/ssh/sshd_config >/dev/null; then
   echo "SSH already disallows root/shell login -- will not modify again."
else
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  / 2>/dev/null
   fi
   echo "PermitRootLogin no"                       >  /ro/etc/ssh/sshd_config
   echo "Banner   /etc/motd"                       >> /ro/etc/ssh/sshd_config
   echo "Subsystem  sftp  /usr/libexec/sftp-server" >> /ro/etc/ssh/sshd_config
```

```
   cp -p /ro/etc/ssh/sshd_config /etc/ssh/
                # sample banner
   echo "WARNING               WARNING               WARNING" >  /ro/etc/motd
   echo ""                                                     >> /ro/etc/motd
   echo " You are accessing an information system that"        >> /ro/etc/motd
   echo " is for authorized users only.  If you are not"       >> /ro/etc/motd
   echo " authorized, log off now."                            >> /ro/etc/motd
   echo ""                                                     >> /ro/etc/motd
   echo "WARNING               WARNING               WARNING" >> /ro/etc/motd
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  / 2>/dev/null
   fi
   /bin/kill -HUP ‘head -1 /var/run/sshd.pid‘
fi
```

6.4. NTP (Network Time Protocol)

Figure 6-4 shows one way of setting up NTP on a Freeway.

Figure 6-4. Enabling NTP

```
export NTP_SERVER=192.168.1.1
export NTP_SERVERB=192.168.1.2

## to synchronize with an NTP (Network Time Protocol) timeserver at powerup
if [ -n "${NTP_SERVER}" ] ; then
   ## /usr/bin2/ntpdate ${NTP_SERVER}
                                    ## to create an NTP configuration file
   echo "server ${NTP_SERVER} prefer"        > /tmp/ntp.conf
   if [ -n "${NTP_SERVERB}" ] ; then
      echo "server ${NTP_SERVERB}"          >> /tmp/ntp.conf
   fi
   echo "driftfile /var/run/ntpd.driftfile" >> /tmp/ntp.conf

                       ## to start an ntpd daemon (see "man ntpd" for details)
   /usr/bin2/ntpd -g -p /tmp/ntpd.pid -c /tmp/ntp.conf
fi
```

6.5. SNMP (Simple Network Management Protocol)

SNMP (Simple Network Management Protocol) can serve as a useful way of checking a Freeway server across the network, but if you don't use it, it should be disabled. Figure 6-5 shows how to disable SNMP.

Figure 6-5. Disabling SNMP

```
                    # For security: prevent snmp from running on this Freeway
if [ -x /usr/local/sbin/snmpd ]; then
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
   chmod ugo-x /usr/local/sbin/snmpd
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi
```

6.6. Secure the Webserver

Figure 6-6 shows some adjustments to make the Freeway webserver more secure. These changes mostly change the configuration of the webserver to cause it to divulge less information about itself and about the Freeway server. Of course, if you don't use the webserver, it is more secure not to enable it so it doesn't run at all.

Figure 6-6. Webserver Security Enhancements

```
         # For security:
         # Add "ServerTokens Prod" line to Apache config, if not already there
export AP_SCMD1="‘sed -e \"/^[Ss][Ee][Rr][Vv][Ee][Rr][Tt][Oo][Kk][Ee][Nn][Ss] *[Pp][Rr][Oo][Dd]/!d\" \
                   /usr/local/etc/apache22/httpd.conf |sed -e \"2,//d\"‘"
if [ "${AP_SCMD1}X" = "X" ]; then
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
   echo "ServerTokens Prod" >> /usr/local/etc/apache22/httpd.conf
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi

                   # For security:
                   # Change any "ServerSignature On" line in Apache config to
                   # "ServerSignature Off"
export AP_SCMD2="‘sed -e \"/^[Ss][Ee][Rr][Vv][Ee][Rr][Ss][Ii][Gg][Nn][Aa][Tt][Uu][Rr][Ee] *[Oo][Nn]/!d\" \
                    /usr/local/etc/apache22/httpd.conf |sed -e \"2,//d\"‘"
if [ ! "${AP_SCMD2}X" = "X" ]; then
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
   mv /usr/local/etc/apache22/httpd.conf /usr/local/etc/apache22/httpd.conf.prev
   sed -e \
      "s/^[Ss][Ee][Rr][Vv][Ee][Rr][Ss][Ii][Gg][Nn][Aa][Tt][Uu][Rr][Ee] *[Oo][Nn]/ServerSignature Off/g" \
       /usr/local/etc/apache22/httpd.conf.prev > /usr/local/etc/apache22/httpd.conf
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi
```

```
          # For security:
          # Add "TraceEnable Off" line to Apache config, if not already there
export AP_SCMD3="‘sed -e \"/^[Tt][Rr][Aa][Cc][Ee][Ee][Nn][Aa][Bb][Ll][Ee] *[Oo][Ff][Ff]/!d\" \
                  /usr/local/etc/apache22/httpd.conf |sed -e \"2,//d\"‘"
if [ "${AP_SCMD3}X" = "X" ]; then
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
   echo "TraceEnable Off" >> /usr/local/etc/apache22/httpd.conf
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi

if [ ! -f /sbin/shutdown.wheel ]; then
   if [ -f /read_only_mounts ]; then
      mount -u -o rw / 2>/dev/null
   fi
   cp -p /sbin/shutdown /sbin/shutdown.wheel
   chgrp wheel /sbin/shutdown.wheel
   if [ -f /read_only_mounts ]; then
      mount -u -o ro / 2>/dev/null
   fi
fi
```

6.7. Enable Logging

Figure 6-7 shows how to configure and enable the `syslogd` daemon. See Chapter 3 for more details.

Figure 6-7. Configure and Enable the syslogd Daemon

```
export LOG_DIR="/var/log"
touch ${LOG_DIR}/all.log
touch ${LOG_DIR}/sra_err.log
touch ${LOG_DIR}/sra_notice.log
touch ${LOG_DIR}/sra_all.log
touch ${LOG_DIR}/sraweb_err.log
touch ${LOG_DIR}/sraweb_all.log

echo "*.*           ${LOG_DIR}/all.log"         > /etc/syslog.conf
echo "local0.err    ${LOG_DIR}/sra_err.log"    >> /etc/syslog.conf
echo "local0.notice ${LOG_DIR}/sra_notice.log" >> /etc/syslog.conf
echo "local0.*      ${LOG_DIR}/sra_all.log"    >> /etc/syslog.conf
echo "local1.err    ${LOG_DIR}/sraweb_err.log" >> /etc/syslog.conf
echo "local1.*      ${LOG_DIR}/sraweb_all.log" >> /etc/syslog.conf

if [ -x /usr/sbin/syslogd ]; then
   /usr/sbin/syslogd
fi
```

6.8. Rotate Log Files

Figure 6-8 shows a method for rotating and maintaining the system logs, including the webserver logs. This method uses the `/usr/sbin/cron` daemon to periodically check the log files, and rotate them if necessary. See Chapter 3 for more details.

Figure 6-8. Rotate the Log Files

```
echo "#!/bin/sh"                                          > /tmp/httplog_rotate.sh
echo "#"                                                 >> /tmp/httplog_rotate.sh
echo "# This script rotates the webserver log files."    >> /tmp/httplog_rotate.sh
echo "# It is expected to be run once per day."          >> /tmp/httplog_rotate.sh
echo "#"                                                 >> /tmp/httplog_rotate.sh
echo "export B_FWY_SERVERNAME=${B_FWY_SERVERNAME}"       >> /tmp/httplog_rotate.sh
echo "#"                                                 >> /tmp/httplog_rotate.sh
echo "rm -f /var/log/httpd-access.log.31.gz"             >> /tmp/httplog_rotate.sh
echo "rm -f /var/log/httpsd-access.log.31.gz"            >> /tmp/httplog_rotate.sh
echo "rm -f /var/log/httpd-error.log.31.gz"              >> /tmp/httplog_rotate.sh
echo "rm -f /var/log/httpsd-error.log.31.gz"             >> /tmp/httplog_rotate.sh
echo "for i in 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 ; do" \
                                                         >> /tmp/httplog_rotate.sh
echo "   export NEWNUM=\"\`expr \${i} + 1\`\""           >> /tmp/httplog_rotate.sh
echo "   mv -f /var/log/httpd-access.log.\${i}.gz   /var/log/httpd-access.log.\${NEWNUM}.gz"  \
                                                         >> /tmp/httplog_rotate.sh
echo "   mv -f /var/log/httpsd-access.log.\${i}.gz  /var/log/httpsd-access.log.\${NEWNUM}.gz" \
                                                         >> /tmp/httplog_rotate.sh
echo "   mv -f /var/log/httpd-error.log.\${i}.gz    /var/log/httpd-error.log.\${NEWNUM}.gz"   \
                                                         >> /tmp/httplog_rotate.sh
echo "   mv -f /var/log/httpsd-error.log.\${i}.gz   /var/log/httpsd-error.log.\${NEWNUM}.gz"  \
                                                         >> /tmp/httplog_rotate.sh
echo "done"                                              >> /tmp/httplog_rotate.sh
echo "/usr/local/etc/rc.d/apache22 stop"                 >> /tmp/httplog_rotate.sh
echo "mv -f /var/log/httpd-access.log  /var/log/httpd-access.log.0"  >> /tmp/httplog_rotate.sh
echo "mv -f /var/log/httpsd-access.log /var/log/httpsd-access.log.0" >> /tmp/httplog_rotate.sh
echo "mv -f /var/log/httpd-error.log   /var/log/httpd-error.log.0"   >> /tmp/httplog_rotate.sh
echo "mv -f /var/log/httpsd-error.log  /var/log/httpsd-error.log.0"  >> /tmp/httplog_rotate.sh
echo "/usr/local/etc/rc.d/apache22 start"                >> /tmp/httplog_rotate.sh
echo "/sbin/gzip /var/log/httpd-access.log.0"            >> /tmp/httplog_rotate.sh
echo "/sbin/gzip /var/log/httpsd-access.log.0"           >> /tmp/httplog_rotate.sh
echo "/sbin/gzip /var/log/httpd-error.log.0"             >> /tmp/httplog_rotate.sh
echo "/sbin/gzip /var/log/httpsd-error.log.0"            >> /tmp/httplog_rotate.sh

chmod go-wx  /tmp/httplog_rotate.sh
chflags schg /tmp/httplog_rotate.sh


echo "# logfilename             mode count size  when flags"    > /etc/newsyslog.conf
echo "${LOG_DIR}/all.log           600  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/cron              600  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/sra_err.log       644  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/sra_notice.log    644  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/sra_all.log       644  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/sraweb_err.log    644  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/sraweb_all.log    644  31     *   @T05  WZ"   >> /etc/newsyslog.conf
```

```
echo "SHELL=/bin/sh"                                          > /etc/crontab
echo "PATH=/etc:/bin:/sbin:/usr/bin:/usr/sbin"               >> /etc/crontab
echo "HOME=/var/log"                                         >> /etc/crontab
echo "B_FWY_SERVERNAME=${B_FWY_SERVERNAME}"                  >> /etc/crontab
echo "MAILTO=\"\""                                           >> /etc/crontab
echo "TZ=\"\""                                               >> /etc/crontab
echo "#"                                                     >> /etc/crontab
echo "#minute hour   mday   month   wday   who     command"   >> /etc/crontab
echo "#"                                                     >> /etc/crontab
echo "0        *      *      *       *      root   newsyslog" >> /etc/crontab
echo "0        5      *      *       *      root   sh /tmp/httplog_rotate.sh" >> /etc/crontab


if [ -x /usr/sbin/cron ]; then
   /usr/sbin/cron
fi
```

6.9. Configure Auditing

Figure 6-9 shows a simple way to setup and enable system-call auditing.

Figure 6-9. Configure Auditing

```
           # Start kernel-level event auditing.  The root user can use
           # "praudit -l /var/audit/current" to see the audit entries, or
           # "praudit -l /dev/auditpipe" to continually see the latest entries
           # as they appear.

if [ ! -d /var/audit ]; then
   mkdir -p -m 750 /var/audit
fi
chmod go-w /etc/security
if /usr/bin/grep "^host:" /etc/security/audit_control >/dev/null; then
   echo "host line already in audit file -- will not tamper with it..."
else
   echo "host:${B_FWY_SERVERNAME}"  >> /etc/security/audit_control
fi

                  # If audit_user file has not been altered by any user, then
                  # add default settings for the 2 initial login accounts.
if [ 5 = ‘cat /etc/security/audit_user |wc -l‘ ]; then
  echo "#"                                                                  >> /etc/security/audit_user
  echo "# These lines have been added to this file by the"                  >> /etc/security/audit_user
  echo "# /usr/local/freeway/boot.src/rc.startsra command script,"          >> /etc/security/audit_user
  echo "# to configure auditing of the 2 originally-configured"            >> /etc/security/audit_user
  echo "# Freeway Monitor users.  To alter these settings, you should add"  >> /etc/security/audit_user
  echo "# echo statements in /usr/local/freeway/boot.src/rc.startsra.local," >> /etc/security/audit_user
  echo "# rather than edit either /etc/security/audit_user or"              >> /etc/security/audit_user
  echo "# /ro/etc/security/audit_user directly; using echo statements"      >> /etc/security/audit_user
  echo "# will ensure that your changes are not lost or altered"            >> /etc/security/audit_user
  echo "# by any subsequent Freeway or Monitor software install."           >> /etc/security/audit_user
```

```
  echo "# See the echo statements near the end of"                          >> /etc/security/audit_user
  echo "# /usr/local/freeway/boot.src/rc.startsra for examples."            >> /etc/security/audit_user
  echo "#"                                                                 >> /etc/security/audit_user
  echo "# All users which should be audited must be added here."            >> /etc/security/audit_user
  echo "#"                                                                 >> /etc/security/audit_user
  echo "user:ex,ap,aa,lo,ad,na,fm,fd,fc,fw,-fr:no"                         >> /etc/security/audit_user
  echo "freeway:ex,ap,aa,lo,ad,na,fm,fd,fc,fw,-fr:no"                      >> /etc/security/audit_user
  echo "#"                                                                 >> /etc/security/audit_user
fi

  # For a description of the format of the /etc/security/audit_user file,
  # run "man audit_user".  For the available event types, see the
  # /etc/security/audit_class and /etc/security/audit_event files.


            # Here is an example of how the audit_warn file could be used to
            # zip and copy each audit trail file when it becomes full.
            # This is commented out, but could be copied to rc.startsra.local
            # and uncommented.

# if [ 6 = `cat /etc/security/audit_warn |wc -l` ]; then
#
#   echo "#"                                                    >> /etc/security/audit_warn
#   echo "# Added by /usr/local/freeway/boot.src/rc.startsra:"  >> /etc/security/audit_warn
#   echo "#"                                                    >> /etc/security/audit_warn
#   echo "# Compress and move audit trail files when they are full." >> /etc/security/audit_warn
#   echo "#"                                                    >> /etc/security/audit_warn
#   echo "export DATEDIR=\"\`date -u -v '-5S' '+%Y%m%d'\`\""    >> /etc/security/audit_warn
#   echo "if [ \"\$1\" = closefile ]; then"                     >> /etc/security/audit_warn
#   echo "   /usr/bin/touch         /var/save/\${DATEDIR}.audit_records.zip" >> /etc/security/audit_warn
#   echo "   /sbin/chown root:audit /var/save/\${DATEDIR}.audit_records.zip" >> /etc/security/audit_warn
#   echo "   /sbin/chmod 600        /var/save/\${DATEDIR}.audit_records.zip" >> /etc/security/audit_warn
#   echo "   /usr/local/bin/zip -r  /var/save/\${DATEDIR}.audit_records.zip \$2" \
                                                              >> /etc/security/audit_warn
##  echo "   /usr/bin/touch         \$2.txt"                   >> /etc/security/audit_warn
##  echo "   /sbin/chown root:audit \$2.txt"                   >> /etc/security/audit_warn
##  echo "   /sbin/chmod 600        \$2.txt"                   >> /etc/security/audit_warn
##  echo "   /usr/sbin/praudit -d '|' \$2 > \$2.txt"           >> /etc/security/audit_warn
##  echo "   /sbin/chmod 400        \$2.txt"                   >> /etc/security/audit_warn
##  echo "   /usr/local/bin/zip -r  /var/save/\${DATEDIR}.audit_records.zip \$2.txt" \
                                                              >> /etc/security/audit_warn
##  echo "   /sbin/rm -f \$2.txt"                              >> /etc/security/audit_warn
#   echo "fi"                                                  >> /etc/security/audit_warn
#
# fi

            # Start the kernel-level audit daemon.
/usr/sbin/auditd

            # Add posixrules file to prevent creating unnecessary audit records
if [ -f /usr/share/zoneinfo/posixrules ]; then
   echo "posixrules file exists."
else
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
   mkdir /usr/share/zoneinfo
```

```
   chmod 755 /usr/share/zoneinfo
   touch /usr/share/zoneinfo/posixrules
   chmod 444 /usr/share/zoneinfo/posixrules
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi

          # Add line to force close/reset of audit file now and every midnight
/usr/sbin/audit -n
if /usr/bin/grep -- "^[^#]*bin\/audit -n" /etc/crontab >/dev/null; then
   echo "Audit file refresh command exists -- will not add again..."
else
   echo "0        0      *      *       *      root   /usr/sbin/audit -n" >> /etc/crontab
fi
```

Chapter 7. Notes

This chapter contains general information to aid in understanding this document.

Table 7-1. Acronym definitions

Acronym	Definition
ICP	Intelligent Communication Processor
IP	Internet Protocol
NTP	Network Time Protocol
SSAOD	System Security Administrator Operators Documentation (SSAOD)
SFUG	Security Features User's Guide
SSH	Secure Shell
TCP/IP	Transmission Control Protocol / Internet Protocol
UDP	User Datagram Protocol / Internet Protocol
WAN	Wide Area Network

Appendix A. Sample rc.startsra File

This appendix shows a sample `rc.startsra` file which configures and enables several of the security-tightening capabililities which are described in other parts of this document.

Note that some of the command lines in this example are broken by "backslash-return" characters at the end of one line, followed by the remainder of the command on the next line. Those commands have been broken to fit into this document, and if copied to a script on a Freeway, either the backslash should be preserved just before the carriage-return at the end of the first line, or the two lines should be concatenated into a single line without the backslash.

A.1. `rc.startsra` Configuration File

Figure A-1. `rc.startsra` File

```
##------------- beginning of rc.startsra file --------------------------------
#!/bin/sh
#
export TZ="GMT"

#
# Additional commands for a Freeway system
# (beyond ordinary Freeway settings and programs)
# This file assumes it is running on a Freeway with a /var/ filesystem
# which is mounted "read-write", rather than the "read-only" which is
# common for Freeways with Flash disks.
#
# The first "export" lines below are intended to be customized
# for each environment:
#
#   DEFAULT_ROUTER is the default IP route.
#
#   The IP addresses specified in TARGET1 and TARGET2 will be used
#   by the ipfailover.sh script to detect when an ethernet interface has
#   failed or become disconnected, and the Freeway should switch
#   to the other ethernet interface.  If ethernet failover is not desired,
#   leave these commented out or blank.  If only one IP can be specified,
#   set both TARGET1 and TARGET2 to that IP address.  If IP addresses
#   are specified, be sure at least one of them is always available for
#   the Freeway to ping; otherwise the Freeway will switch between
#   the two ethernet interfaces every 70 seconds or so.
#
#   NTP_SERVER and NTP_SERVERB are expected to be NTP servers, to which
#   the Freeway will synchronize its time (if only one NTP server
#   is available, leave NTP_SERVERB blank; if none available, leave
#   NTP_SERVER blank).
#

echo "No logins permitted until auditing has started." > /var/run/nologin

# export DEFAULT_ROUTER=192.168.1.3
# export TARGET1=192.168.1.3
```

```
# export TARGET2=192.168.1.2
export NTP_SERVER=192.168.1.1
export NTP_SERVERB=192.168.1.2

             # setup to write any core files to the read-write /var/ partition
sysctl kern.corefile=/var/%N_%P.core

# sysctl net.inet.tcp.keepidle=300000
# sysctl net.inet.tcp.keepintvl=60000
# sysctl net.inet.tcp.always_keepalive=1

                              # For security: disable all pkt forwarding,
                              # using both sysctl and the firewall;
                              # disallow sending or receiving of any packet
                              # which is not sourced or destined directly
                              # to this Freeway;
                              # also disallow sending of ICMP "time exceeded"
                              # packets, which could be used by traceroute
                              # to discover information about the network.
sysctl net.inet.ip.forwarding=0
sysctl net.inet6.ip6.forwarding=0
ipfw add 20100 deny ip from not me to not me
ipfw add 20200 deny icmp from me to any icmptypes 11

            # For security: deny all access to timestamp info via ICMP packets
ipfw add 20300 deny icmp from any to any icmptypes 13
ipfw add 20400 deny icmp from any to any icmptypes 14

            # For security: deny all access to unnecessary ports
ipfw add 21000 deny tcp from any to me 23    # telnet
ipfw add 21100 deny tcp from any to me 80    # http
ipfw add 21200 deny tcp from any to me 513   # login
ipfw add 21300 deny tcp from any to me 514   # shell
# ipfw add 21400 deny tcp from any to me 20    # ftp-data
# ipfw add 21500 deny tcp from any to me 21    # ftp
# ipfw add 21600 deny tcp from any to me 8208  # Freeway daemon

            # For security: Turn telnet and rlogin off in inetd.conf, too
if [ -f /read_only_mounts ]; then
   mount -u -o rw  / 2>/dev/null
fi
mv /ro/etc/inetd.conf /ro/etc/inetd.conf.prev
sed -e "s/^t/#t/g" /ro/etc/inetd.conf.prev |sed -e "s/^l/#l/g" > /ro/etc/inetd.conf
if [ -f /read_only_mounts ]; then
   mount -u -o ro  / 2>/dev/null
fi

              # For security: Disallow direct root or shell login via ssh
if /usr/bin/grep -- "^[^#]*PermitRootLogin no" /ro/etc/ssh/sshd_config >/dev/null; then
   echo "SSH already disallows root/shell login -- will not modify again."
else
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  / 2>/dev/null
   fi
   echo "PermitRootLogin no"                      >  /ro/etc/ssh/sshd_config
   echo "Banner   /etc/motd"                      >> /ro/etc/ssh/sshd_config
   echo "Subsystem  sftp  /usr/libexec/sftp-server" >> /ro/etc/ssh/sshd_config
```

```
   cp -p /ro/etc/ssh/sshd_config /etc/ssh/
                # sample banner
   echo "WARNING              WARNING              WARNING" >  /ro/etc/motd
   echo ""                                                   >> /ro/etc/motd
   echo " You are accessing an information system that"      >> /ro/etc/motd
   echo " is for authorized users only.  If you are not"     >> /ro/etc/motd
   echo " authorized, you must log off now."                 >> /ro/etc/motd
   echo ""                                                   >> /ro/etc/motd
   echo "WARNING              WARNING              WARNING" >> /ro/etc/motd
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  / 2>/dev/null
   fi
   /bin/kill -HUP `head -1 /var/run/sshd.pid`
fi

       # Allow local configuration overrides (rc.startsra.local can be created
       # by customers to customize a specific Freeway, without the risk of
       # being overwritten by the next software upgrade -- because software
       # upgrades will not overwrite any rc.startsra.local* file).

if [ -f /tmp/boot/rc.startsra.local ]; then
   . /tmp/boot/rc.startsra.local
fi

## to synchronize with an NTP (Network Time Protocol) timeserver at powerup
if [ -n "${NTP_SERVER}" ] ; then
   ## /usr/bin2/ntpdate ${NTP_SERVER}
                                      ## to create an NTP configuration file
   echo "server ${NTP_SERVER} prefer"        > /tmp/ntp.conf
   if [ -n "${NTP_SERVERB}" ] ; then
      echo "server ${NTP_SERVERB}"          >> /tmp/ntp.conf
   fi
   echo "driftfile /var/run/ntpd.driftfile" >> /tmp/ntp.conf

                     ## to start an ntpd daemon (see "man ntpd" for details)
   /usr/bin2/ntpd -g -p /tmp/ntp.pid -c /tmp/ntp.conf
fi

                    # For security: prevent snmp from running on this Freeway
if [ -x /usr/local/sbin/snmpd ]; then
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
   chmod ugo-x /usr/local/sbin/snmpd
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi

        # For security:
        # Add "ServerTokens Prod" line to Apache config, if not already there
export AP_SCMD1="`sed -e \"/^[Ss][Ee][Rr][Vv][Ee][Rr][Tt][Oo][Kk][Ee][Nn][Ss] *[Pp][Rr][Oo][Dd]/!d\" \
                  /usr/local/etc/apache22/httpd.conf |sed -e \"2,//d\"`"
if [ "${AP_SCMD1}X" = "X" ]; then
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
```

```
   echo "ServerTokens Prod" >> /usr/local/etc/apache22/httpd.conf
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi

                  # For security:
                  # Change any "ServerSignature On" line in Apache config to
                  # "ServerSignature Off"
export AP_SCMD2="‘sed -e \"/^[Ss][Ee][Rr][Vv][Ee][Rr][Ss][Ii][Gg][Nn][Aa][Tt][Uu][Rr][Ee] *[Oo][Nn]/!d\" \
                   /usr/local/etc/apache22/httpd.conf |sed -e \"2,//d\"‘"
if [ ! "${AP_SCMD2}X" = "X" ]; then
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
   mv /usr/local/etc/apache22/httpd.conf /usr/local/etc/apache22/httpd.conf.prev
   sed -e \
      "s/^[Ss][Ee][Rr][Vv][Ee][Rr][Ss][Ii][Gg][Nn][Aa][Tt][Uu][Rr][Ee] *[Oo][Nn]/ServerSignature Off/g" \
       /usr/local/etc/apache22/httpd.conf.prev > /usr/local/etc/apache22/httpd.conf
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi

          # For security:
          # Add "TraceEnable Off" line to Apache config, if not already there
export AP_SCMD3="‘sed -e \"/^[Tt][Rr][Aa][Cc][Ee][Ee][Nn][Aa][Bb][Ll][Ee] *[Oo][Ff][Ff]/!d\" \
                  /usr/local/etc/apache22/httpd.conf |sed -e \"2,//d\"‘"
if [ "${AP_SCMD3}X" = "X" ]; then
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
   echo "TraceEnable Off" >> /usr/local/etc/apache22/httpd.conf
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi

if [ ! -f /sbin/shutdown.wheel ]; then
   if [ -f /read_only_mounts ]; then
      mount -u -o rw / 2>/dev/null
   fi
   cp -p /sbin/shutdown /sbin/shutdown.wheel
   chgrp wheel /sbin/shutdown.wheel
   if [ -f /read_only_mounts ]; then
      mount -u -o ro / 2>/dev/null
   fi
fi

export LOG_DIR="/var/log"
touch ${LOG_DIR}/all.log
touch ${LOG_DIR}/sra_err.log
touch ${LOG_DIR}/sra_notice.log
touch ${LOG_DIR}/sra_all.log
touch ${LOG_DIR}/sraweb_err.log
touch ${LOG_DIR}/sraweb_all.log
```

```
echo "*.*           ${LOG_DIR}/all.log"         > /etc/syslog.conf
echo "local0.err    ${LOG_DIR}/sra_err.log"    >> /etc/syslog.conf
echo "local0.notice ${LOG_DIR}/sra_notice.log" >> /etc/syslog.conf
echo "local0.*      ${LOG_DIR}/sra_all.log"    >> /etc/syslog.conf
echo "local1.err    ${LOG_DIR}/sraweb_err.log" >> /etc/syslog.conf
echo "local1.*      ${LOG_DIR}/sraweb_all.log" >> /etc/syslog.conf

if [ -x /usr/sbin/syslogd ]; then
   /usr/sbin/syslogd
fi


echo "#!/bin/sh"                                          > /tmp/httplog_rotate.sh
echo "#"                                                 >> /tmp/httplog_rotate.sh
echo "# This script rotates the webserver log files."    >> /tmp/httplog_rotate.sh
echo "# It is expected to be run once per day."          >> /tmp/httplog_rotate.sh
echo "#"                                                 >> /tmp/httplog_rotate.sh
echo "export B_FWY_SERVERNAME=${B_FWY_SERVERNAME}"       >> /tmp/httplog_rotate.sh
echo "#"                                                 >> /tmp/httplog_rotate.sh
echo "rm -f /var/log/httpd-access.log.31.gz"             >> /tmp/httplog_rotate.sh
echo "rm -f /var/log/httpsd-access.log.31.gz"            >> /tmp/httplog_rotate.sh
echo "rm -f /var/log/httpd-error.log.31.gz"              >> /tmp/httplog_rotate.sh
echo "rm -f /var/log/httpsd-error.log.31.gz"             >> /tmp/httplog_rotate.sh
echo "for i in 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 ; do" \
                                                         >> /tmp/httplog_rotate.sh
echo "   export NEWNUM=\"\`expr \${i} + 1\`\""           >> /tmp/httplog_rotate.sh
echo "   mv -f /var/log/httpd-access.log.\${i}.gz   /var/log/httpd-access.log.\${NEWNUM}.gz"  \
                                                         >> /tmp/httplog_rotate.sh
echo "   mv -f /var/log/httpsd-access.log.\${i}.gz  /var/log/httpsd-access.log.\${NEWNUM}.gz" \
                                                         >> /tmp/httplog_rotate.sh
echo "   mv -f /var/log/httpd-error.log.\${i}.gz    /var/log/httpd-error.log.\${NEWNUM}.gz"   \
                                                         >> /tmp/httplog_rotate.sh
echo "   mv -f /var/log/httpsd-error.log.\${i}.gz   /var/log/httpsd-error.log.\${NEWNUM}.gz"  \
                                                         >> /tmp/httplog_rotate.sh
echo "done"                                              >> /tmp/httplog_rotate.sh
echo "/usr/local/etc/rc.d/apache22 stop"                 >> /tmp/httplog_rotate.sh
echo "mv -f /var/log/httpd-access.log  /var/log/httpd-access.log.0"  >> /tmp/httplog_rotate.sh
echo "mv -f /var/log/httpsd-access.log /var/log/httpsd-access.log.0" >> /tmp/httplog_rotate.sh
echo "mv -f /var/log/httpd-error.log   /var/log/httpd-error.log.0"   >> /tmp/httplog_rotate.sh
echo "mv -f /var/log/httpsd-error.log  /var/log/httpsd-error.log.0"  >> /tmp/httplog_rotate.sh
echo "/usr/local/etc/rc.d/apache22 start"                >> /tmp/httplog_rotate.sh
echo "/sbin/gzip /var/log/httpd-access.log.0"            >> /tmp/httplog_rotate.sh
echo "/sbin/gzip /var/log/httpsd-access.log.0"           >> /tmp/httplog_rotate.sh
echo "/sbin/gzip /var/log/httpd-error.log.0"             >> /tmp/httplog_rotate.sh
echo "/sbin/gzip /var/log/httpsd-error.log.0"            >> /tmp/httplog_rotate.sh

chmod go-wx  /tmp/httplog_rotate.sh
chflags schg /tmp/httplog_rotate.sh


echo "# logfilename             mode count size  when flags"    > /etc/newsyslog.conf
echo "${LOG_DIR}/all.log           600  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/cron              600  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/sra_err.log       644  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/sra_notice.log    644  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/sra_all.log       644  31     *   @T05  WZ"   >> /etc/newsyslog.conf
echo "${LOG_DIR}/sraweb_err.log    644  31     *   @T05  WZ"   >> /etc/newsyslog.conf
```

```
echo "${LOG_DIR}/sraweb_all.log    644  31     *   @T05  WZ"   >> /etc/newsyslog.conf


echo "SHELL=/bin/sh"                                            > /etc/crontab
echo "PATH=/etc:/bin:/sbin:/usr/bin:/usr/sbin"                 >> /etc/crontab
echo "HOME=/var/log"                                           >> /etc/crontab
echo "B_FWY_SERVERNAME=${B_FWY_SERVERNAME}"                    >> /etc/crontab
echo "MAILTO=\"\""                                             >> /etc/crontab
echo "TZ=\"\""                                                 >> /etc/crontab
echo "#"                                                       >> /etc/crontab
echo "#minute hour   mday   month   wday   who     command"   >> /etc/crontab
echo "#"                                                       >> /etc/crontab
echo "0        *      *      *       *      root   newsyslog" >> /etc/crontab
echo "0        5      *      *       *      root   sh /tmp/httplog_rotate.sh" >> /etc/crontab


if [ -x /usr/sbin/cron ]; then
   /usr/sbin/cron
fi

# figure out which type Ethernets we have, fxp0/fxp1 (old) or em0/em1
export ETH_DEV="'ifconfig |sed -e '2,/*/d' |sed -e 's/0.*//''"

if [ -n ${ETH_DEV} -a -n "${TARGET1}" -a -n "${TARGET2}" ] ; then

echo "#!/bin/sh"                                                 > /tmp/ipfailover.sh
echo "#"                                                        >> /tmp/ipfailover.sh
echo "# This script alternates between ${ETH_DEV}0 and ${ETH_DEV}1, whenever it detects" \
                                                                >> /tmp/ipfailover.sh
echo "# a failure on the interface which is currently in use." >> /tmp/ipfailover.sh
echo "#"                                                        >> /tmp/ipfailover.sh
echo "if [ -z \"${TARGET1}\" -o -z \"${TARGET2}\" ] ; then" >> /tmp/ipfailover.sh
echo "   exit 1"                                                >> /tmp/ipfailover.sh
echo "fi"                                                       >> /tmp/ipfailover.sh
echo ""                                                         >> /tmp/ipfailover.sh
echo "ifconfig ${ETH_DEV}0 down"                                >> /tmp/ipfailover.sh
echo "ifconfig ${ETH_DEV}1 down"                                >> /tmp/ipfailover.sh
echo "export ETH0_ETHERLINE=\"\'ifconfig ${ETH_DEV}0 | sed \\\"/ether/!d\\\"\'\"" >> /tmp/ipfailover.sh
echo "           # Note: The 2 bracketed areas in the line below each"      >> /tmp/ipfailover.sh
echo "           #      contain one tab character and one space character." >> /tmp/ipfailover.sh
echo "export ETH0_INETLINE=\"\'ifconfig ${ETH_DEV}0 | sed \\\"/[        ]*inet[         ]*/!d\\\" | \
                           sed \\\"2,\\\\\$d\\\"\'\"" >> /tmp/ipfailover.sh
echo "ifconfig ${ETH_DEV}1 \${ETH0_ETHERLINE}"                  >> /tmp/ipfailover.sh
echo "ifconfig ${ETH_DEV}1 \${ETH0_INETLINE}"                   >> /tmp/ipfailover.sh
echo ""                                                         >> /tmp/ipfailover.sh
echo "# do forever"                                             >> /tmp/ipfailover.sh
echo "while true ; do"                                          >> /tmp/ipfailover.sh
echo ""                                                         >> /tmp/ipfailover.sh
echo "   # echo resetting to use ${ETH_DEV}0"                   >> /tmp/ipfailover.sh
echo "   ifconfig ${ETH_DEV}1 down"                             >> /tmp/ipfailover.sh
echo "   ifconfig ${ETH_DEV}0 up"                               >> /tmp/ipfailover.sh
echo "   sleep 30"                                              >> /tmp/ipfailover.sh
echo ""                                                         >> /tmp/ipfailover.sh
echo "      # stay here as long as we can ping either target" >> /tmp/ipfailover.sh
echo "   while ping -n -o -t 10 ${TARGET1} > /dev/null ||"    >> /tmp/ipfailover.sh
echo "         ping -n -o -t 10 ${TARGET2} > /dev/null ||"    >> /tmp/ipfailover.sh
echo "         ping -n -o -t 10 ${TARGET1} > /dev/null ||"    >> /tmp/ipfailover.sh
```

```
echo "         ping -n -o -t 10 ${TARGET2} > /dev/null ; do"  >> /tmp/ipfailover.sh
echo "      sleep 10"                                         >> /tmp/ipfailover.sh
echo "   done"                                                >> /tmp/ipfailover.sh
echo ""                                                      >> /tmp/ipfailover.sh
echo "   # echo resetting to use ${ETH_DEV}1"                 >> /tmp/ipfailover.sh
echo "   ifconfig ${ETH_DEV}0 down"                           >> /tmp/ipfailover.sh
echo "   ifconfig ${ETH_DEV}1 up"                             >> /tmp/ipfailover.sh
echo "   logger -p local0.warning -s \"Ethernet ${ETH_DEV}0 failed, switching to ${ETH_DEV}1\"" \
                                                             >> /tmp/ipfailover.sh
echo "   sleep 30"                                            >> /tmp/ipfailover.sh
echo ""                                                      >> /tmp/ipfailover.sh
echo "      # stay here as long as we can ping either target" >> /tmp/ipfailover.sh
echo "   while ping -n -o -t 10 ${TARGET1} > /dev/null ||"    >> /tmp/ipfailover.sh
echo "         ping -n -o -t 10 ${TARGET2} > /dev/null ||"    >> /tmp/ipfailover.sh
echo "         ping -n -o -t 10 ${TARGET1} > /dev/null ||"    >> /tmp/ipfailover.sh
echo "         ping -n -o -t 10 ${TARGET2} > /dev/null ; do"  >> /tmp/ipfailover.sh
echo "      sleep 10"                                         >> /tmp/ipfailover.sh
echo "   done"                                                >> /tmp/ipfailover.sh
echo "   logger -p local0.warning -s \"Ethernet ${ETH_DEV}1 failed, switching to ${ETH_DEV}0\"" \
                                                             >> /tmp/ipfailover.sh
echo ""                                                      >> /tmp/ipfailover.sh
echo "done"                                                  >> /tmp/ipfailover.sh
echo " "                                                     >> /tmp/ipfailover.sh
chmod go-wx  /tmp/ipfailover.sh
chflags schg /tmp/ipfailover.sh

sh /tmp/ipfailover.sh &

fi

# route delete default
# route add default ${DEFAULT_ROUTER}

cd /tmp/boot


##   Start webserver (commented out)
##if [ -x /usr/local/etc/rc.d/apache22 ]; then
##  /usr/local/etc/rc.d/apache22 start
##fi

if [ ! -d /var/save ]; then
   mkdir -p -m 777 /var/save
fi

ln -s M /etc/malloc.conf

           # Start kernel-level event auditing.  The root user can use
           # "praudit -l /var/audit/current" to see the audit entries, or
           # "praudit -l /dev/auditpipe" to continually see the latest entries
           # as they appear.

if [ ! -d /var/audit ]; then
   mkdir -p -m 750 /var/audit
fi
chmod go-w /etc/security
if /usr/bin/grep "^host:" /etc/security/audit_control >/dev/null; then
```

```
   echo "host line already in audit file -- will not tamper with it..."
else
   echo "host:${B_FWY_SERVERNAME}"  >> /etc/security/audit_control
fi

                  # If audit_user file has not been altered by any user, then
                  # add default settings for the 2 initial login accounts.
if [ 5 = ‘cat /etc/security/audit_user |wc -l‘ ]; then
   echo "#"                                                              >> /etc/security/audit_user
   echo "# These lines have been added to this file by the"              >> /etc/security/audit_user
   echo "# /usr/local/freeway/boot.src/rc.startsra command script,"     >> /etc/security/audit_user
   echo "# to configure auditing of the 2 originally-configured"        >> /etc/security/audit_user
   echo "# Freeway Monitor users.  To alter these settings, you should add"   >> /etc/security/audit_user
   echo "# echo statements in /usr/local/freeway/boot.src/rc.startsra.local," >> /etc/security/audit_user
   echo "# rather than edit either /etc/security/audit_user or"         >> /etc/security/audit_user
   echo "# /ro/etc/security/audit_user directly; using echo statements" >> /etc/security/audit_user
   echo "# will ensure that your changes are not lost or altered"       >> /etc/security/audit_user
   echo "# by any subsequent Freeway or Monitor software install."      >> /etc/security/audit_user
   echo "# See the echo statements near the end of"                     >> /etc/security/audit_user
   echo "# /usr/local/freeway/boot.src/rc.startsra for examples."       >> /etc/security/audit_user
   echo "#"                                                            >> /etc/security/audit_user
   echo "# All users which should be audited must be added here."       >> /etc/security/audit_user
   echo "#"                                                            >> /etc/security/audit_user
   echo "user:ex,ap,aa,lo,ad,na,fm,fd,fc,fw,-fr:no"                    >> /etc/security/audit_user
   echo "freeway:ex,ap,aa,lo,ad,na,fm,fd,fc,fw,-fr:no"                 >> /etc/security/audit_user
   echo "#"                                                            >> /etc/security/audit_user
fi

   # For a description of the format of the /etc/security/audit_user file,
   # run "man audit_user".  For the available event types, see the
   # /etc/security/audit_class and /etc/security/audit_event files.


            # Here is an example of how the audit_warn file could be used to
            # zip and copy each audit trail file when it becomes full.
            # This is commented out, but could be copied to rc.startsra.local
            # and uncommented.

# if [ 6 = ‘cat /etc/security/audit_warn |wc -l‘ ]; then
#
#   echo "#"                                                    >> /etc/security/audit_warn
#   echo "# Added by /usr/local/freeway/boot.src/rc.startsra:"  >> /etc/security/audit_warn
#   echo "#"                                                    >> /etc/security/audit_warn
#   echo "# Compress and move audit trail files when they are full." >> /etc/security/audit_warn
#   echo "#"                                                    >> /etc/security/audit_warn
#   echo "export DATEDIR=\"\‘date -u -v ’-5S’ ’+%Y%m%d’\‘\""    >> /etc/security/audit_warn
#   echo "if [ \"\$1\" = closefile ]; then"                     >> /etc/security/audit_warn
#   echo "   /usr/bin/touch         /var/save/\${DATEDIR}.audit_records.zip" >> /etc/security/audit_warn
#   echo "   /sbin/chown root:audit /var/save/\${DATEDIR}.audit_records.zip" >> /etc/security/audit_warn
#   echo "   /sbin/chmod 600        /var/save/\${DATEDIR}.audit_records.zip" >> /etc/security/audit_warn
#   echo "   /usr/local/bin/zip -r  /var/save/\${DATEDIR}.audit_records.zip \$2" \
                                                              >> /etc/security/audit_warn
##  echo "   /usr/bin/touch         \$2.txt"                  >> /etc/security/audit_warn
##  echo "   /sbin/chown root:audit \$2.txt"                  >> /etc/security/audit_warn
##  echo "   /sbin/chmod 600        \$2.txt"                  >> /etc/security/audit_warn
##  echo "   /usr/sbin/praudit -d ’|’ \$2 > \$2.txt"          >> /etc/security/audit_warn
##  echo "   /sbin/chmod 400        \$2.txt"                  >> /etc/security/audit_warn
```

```
##  echo "   /usr/local/bin/zip -r  /var/save/\${DATEDIR}.audit_records.zip \$2.txt" \
                                                                    >> /etc/security/audit_warn
##  echo "   /sbin/rm -f \$2.txt"                                   >> /etc/security/audit_warn
#   echo "fi"                                                      >> /etc/security/audit_warn
#
# fi

            # Start the kernel-level audit daemon.
/usr/sbin/auditd

            # Add posixrules file to prevent creating unnecessary audit records
if [ -f /usr/share/zoneinfo/posixrules ]; then
   echo "posixrules file exists."
else
   if [ -f /read_only_mounts ]; then
      mount -u -o rw  /usr 2>/dev/null
   fi
   mkdir /usr/share/zoneinfo
   chmod 755 /usr/share/zoneinfo
   touch /usr/share/zoneinfo/posixrules
   chmod 444 /usr/share/zoneinfo/posixrules
   if [ -f /read_only_mounts ]; then
      mount -u -o ro  /usr 2>/dev/null
   fi
fi

          # Allow final local configuration overrides or additions
          # (adding lines to /var/crontab, for example).
          # rc.startsra.local2 can be created by customers to customize
          # a specific Freeway, just like rc.startsra.local can,
          # without the risk of being overwritten by the next software
          # upgrade -- because software upgrades will not overwrite
          # any rc.startsra.local* file.

if [ -f /tmp/boot/rc.startsra.local2 ]; then
   . /tmp/boot/rc.startsra.local2
fi

          # Add line to force close/reset of audit file now and every midnight
/usr/sbin/audit -n
if /usr/bin/grep -- "^[^#]*bin\/audit -n" /etc/crontab >/dev/null; then
   echo "Audit file refresh command exists -- will not add again..."
else
   echo "0        0      *      *       *      root   /usr/sbin/audit -n" >> /etc/crontab
fi

          # Allow logins
rm -f /var/run/nologin

##---------------- end of rc.startsra file ------------------------------
```

Index

A

Acronyms, 27
  ICP (Intelligent Communications Processor)
    (see ICP)
  IP (Internet Protocol)
    (see IP)
  NTP (Network Time Protocol)
    (see NTP)
  SFUG (Security Features User's Guide)
    (see SFUG)
  SSH (Secure Shell)
    (see SSH)
  TCP/IP (Transmission Control Protocol)
    (see TCP/IP)
  UDP (User Datagram Protocol)
    (see UDP)
  WAN (Wide Area Network)
    (see WAN)
Audience, v
audit, 15
auditing, 24
audit_warn, 16

C

cron, 12, 16
CSCI, 27
Customer support, ix

D

Data Item Description
  (see DID)
DID, 11, 27
Document conventions, viii

F

firewall, 14, 18

I

ICP, 10, 27
Identification, 10
Intelligent Communications Processor
  (see ICP)
Internet Protocol
  (see IP)
IP, 10, 27
ipfw (firewall), 14

L

logging, 22

N

Network Time Protocol
  (see NTP)
newsyslog, 12
Notes, 27
NTP, 20, 27

P

Preface, v
Product support, ix

R

rc.startsra, 12, 14, 15
  rc.startsra, 28
Reference documents, vi, 11
rlogin, 19

S

Secure Shell
(see SSH)
Security Features User's Guide
(see SFUG)
SFUG, 27
SNMP, 20
SSAOD, 27
SSH, 19, 19, 27
Support, product, ix
syslog, 12, 22

T

TCP/IP, 27
Technical support, ix
telnet, 19
Transmission Control Protocol
(see TCP/IP)

U

UDP, 27
unnecessary services, 19
User Datagram Protocol
(see UDP)

W

WAN, 10, 27
Wide Area Network
(see WAN)

Customer Report Form

Customer Report Form

We at Protogate are constantly striving to improve our products. If you have any suggestions or problems you would like to report regarding our hardware, software, or documentation, please complete the following form and mail it to us at Protogate, Inc., 12225 World Trade Drive, Suite R, San Diego, CA, 92128, USA. Or contact us via email: <sales@protogate.com>, voice: (858) 451-0865, or fax: (877) 473-0190. Please also include the document title or number and the section and page number, if applicable.

Your Name and Phone Number:

_____

Company:

_____

Address:

_____

_____

_____

Product:

_____

Problem or Suggestion:

_____

_____

_____

_____

_____

Thank you.